

# QR kód lokalizáció kaszkádolt gyenge osztályozók használatával\*

Bodnár Péter és Nyúl László

Szegedi Tudományegyetem  
Képfeldolgozás és Számítógépes Grafika Tanszék  
{bodnaar, nyul}@inf.u-szeged.hu

**Absztrakt** A számítógéppel automatikusan olvasható vizuális kódok a mindennapjaink részévé váltak az ipari alkalmazásokban és a személyes felhasználási területeken is. A leolvasási folyamat két lépésből áll, a kód lokalizálásából és a hordozott adat dekódolásából. Ipari környezetben vagy csökkent látóképességű felhasználóknál emberi beavatkozást nem igénylő lokalizálás szükséges. Az alábbiakban bemutatjuk a kaszkádolt gyenge osztályozók lokalizálási képességeit, melyek Haar-jellemzőket, Lokális Bináris Mintákat (Local Binary Patterns, LBP) és Gradiens Irány Hisztogramot (Histogram of Oriented Gradients, HOG) használnak. Ezeket az osztályozókat kipróbáljuk a QR kód lokalizációs mintájára tanítva és a teljes kódon tanítva is. Az utófeldolgozás javítására további javaslatokat teszünk.

## 1. Bevezetés

A QR kód egy vizuális kódszabvány, melyet elterjedten használnak ipari alkalmazásokban és magáncélú projektekben egyaránt. A formátumának köszönhetően automatikus adatleolvasást tesz lehetővé számítógépek és beágyazott rendszerek számára (1. ábra). Ezek a kódok nagyobb mennyiségű beágyazott adat hordozására képesek, mint egydimenziós őseik, a vonalkódok. A QR kódok használata jelentős mértékben nőtt az elmúlt években, felülmúlva más kétdimenziós kódszabványok, mint például az Aztec code és a Maxicode fejlődési ütemét. Ez annak is köszönhető, hogy fejlett hibajavítással rendelkezik, ami biztosítja az adat helyreállítását akár 30 % vizuális sérülés esetén is.

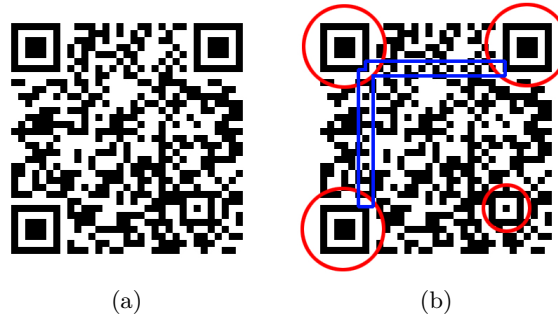
Az elmúlt években a képalkotás fejlődése miatt lehetővé vált a QR kódok való idejű automatikus leolvasása. A jelenlegi algoritmusok nem igényelnek többé emberi beavatkozást a kód leolvasásához, sem feltételezéseket a kód helyzetére és méretére a képtérben [1, 2]. A képalkotó berendezések, a képminőség és a végfelhasználói alkalmazások sokfélesége változatos igényeket támaszt a leolvasás sebességét és pontosságát illetően, ami nehezíti általános célú algoritmusok megtervezését.

---

\* A dolgozat angol nyelven a következő helyen jelent meg: P. Bodnár, L.G. Nyúl, „QR Code Localization Using Boosted Cascade of Weak Classifiers”, Image Analysis and Recognition, 11th International Conference, ICIAR 2014, Proceedings, Part I, LNCS 8814, pp. 338–345, 2014.

A QR kódok visszafejtésének két lépése a kód lokalizálása és a hordozott adat dekódolása. A szakirodalom széles körben tárgyal hatékony QR kód lokalizálási algoritmusokat [3–5], melyek a végfelhasználó igényeinek megfelelően nagyobb sebességre vagy pontosságra optimalizálnak. A morfológiai műveletek, a konvolúció, a sarokdetektálás pontos felismerést tehetnek lehetővé, miközben jelentősen lassítják a rendszert. Az egyszerű algoritmusok ezzel szemben könnyen futtathatók beágyazott rendszereken, pontosságban viszont alulmaradnak.

Belussi [5] QR kód lokalizáló algoritmus a Viola-Jones keretrendszer [6] köré épült, melyről bebizonyította, hogy bár a keretrendszer eredetileg arcfelismeréshez készült, megfelelő a feladatra még alacsony felbontáson is. A szerzők gyenge osztályozók kaszkádját használták, melyeket a kód lokalizációs mintájára (Finder Pattern, FIP) tanítottak. A következő részben kiterjesztjük az eredeti ötletet, megvizsgáljuk más jellemzőkön és más bemenő adatokon tanult kaszkád osztályozók teljesítményét.



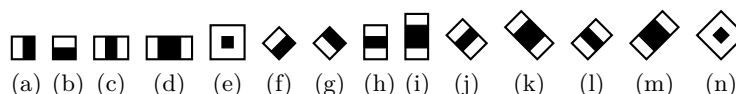
1. ábra. (a): QR kód példa, (b): ugyanaz a kód megjelölt lokalizálási mintákkal (piros kör) és adatsűrűségi mintákkal (kék téglalap)

## 2. Lokalizáció kaszkád osztályozó tanításával

Az általános osztályozási feladatokban gyakran használt módszer a gyenge osztályozók használata és hatékonyságuk javítása Boosting segítségével. Egy gyenge osztályozó gyorsan tanítható, de az önmagában vett hatékonysága gyenge. Ezt szokás még a gyenge osztályozók egymás után fűzésével áthidalni. Az eljárás során egy gyenge osztályozó, ami tipikusan 0.99 körüli TPR (True Positive Rate) és 0.5 körüli FPR (False Positive Rate) értékkel rendelkezik, átadja a kimenetét a következő osztályozó bemenetére. Az együttes hatékonyság így  $TPR^n$ , ahol  $n$  a szintek száma. A módszer segítségével képesek vagyunk egy összetett problémára egyszerű jellemzőkön tanított gyenge osztályozókat felhasználva egy erős kaszkád osztályozót adni.

## 2.1. Kinyert jellemzők

Viola és Jones [6] mutatta be a Haar jellemzők használatát a képfeldolgozásban, melyek néhány gyenge osztályozónkat és a referencia-osztályozót alkotják. Ezek a jellemzők a nevüket a Haar wavelet függvénycsalád függvényeiről kapták, és három csoportra oszthatóak: él-típusú, vonal-típusú és középpont-típusú jellemzőkre (2. ábra). Ezek a halmazok kiterjeszthetők a 45 fokban elforgatott jellemzőkkel Lienhart javaslatára [7].



2. ábra. Haar jellemzők: él típusú (a, b), vonal típusú (c, d), középpont típusú (e), és elforgatott entitások (f-n)

Minden osztályozó egy vagy több jellemzőt tartalmaz a fenti jellemzőcsaládból. A Haar jellemzőket definiálja a függvényalak, a skálázás mértéke és az elhelyezkedése az ablakozott képrészleten (ROI) belül. Az osztályozás folyamata a gyenge osztályozók kaszkádjának csúszóablakos kiértékelése. Ez a lépés megismételhető tetszőleges skálázással, így a kötött méreten tanult kaszkád osztályozó képes felismerni a tanítópéldáknál nagyobb méretű objektumokat is. A feldolgozás eredménye azon befoglaló dobozok halmaza, melyet az osztályozó megtalált. Ezeket szokás tovább szűrni annak függvényében, hogy egy adott területre adott átfedési toleranciával hány szavazat érkezett. Az egy területre eső jelöltek összevonhatók, míg a kevés szavazatot kapott területek elvethetők. A jellemzők kiértékelése tovább gyorsítható integrálképek használatával, melyek lehetővé teszik a valós idejű, multi-scale feldolgozást.

Haar jellemzők használata helyett Lokális Bináris Minták (Local Binary Patterns, LBP) és Gradiens Irány Histogrammot (Histogram of Oriented Gradients, HOG) is használhatunk az osztályozók tanítására. Egy korábbi munka [8] a kép uniform partícionálásával és a blokkon belül kör mentén beolvasott pixelek egydimenziós vektorának feldolgozásával nyer ki jellemzőket és dönti el a vizuális kód valószínűségét a blokkon belül. Ez az elv analóg az LBP-vel [9], annyi különbséggel, hogy a középpontnak nem ad kitüntetett szerepet. Pixelek feldolgozása helyett az LBP tetszés szerinti blokkokkal és azok átlagos értékeivel is képes dolgozni.

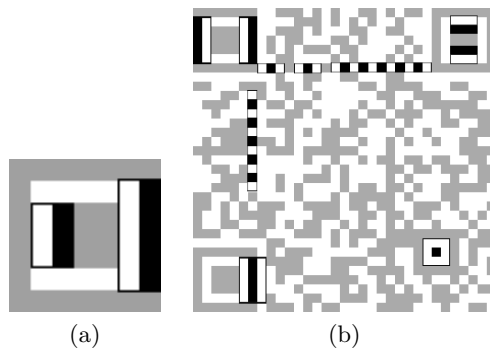
A HOG jellemzők először gyalogosok detektálási feladatában jelentek meg [10], de gyakran vizsgálják a használatát a jellemzőkinyerés és a számítógépes látás olyan problémáinál, ahol LBP vagy SIFT használható. Vannak speciális esetek is [11], ahol LBP és HOG együttes használata növeli az algoritmus hatékonyságát.

## 2.2. Lokalizáció a FIP-ekre és az egész kódra alapozva

Belussi munkája [5] adja a fejlesztéseink alapját. A szerző alapötlete Haar jellemzők tanítása a QR kód három sarkában található fő FIP-ekre magas recall (hit rate) értékkel, aztán a FIP-jelöltek hármasokba szervezhetők, melyek lehetséges QR kód-területeket definiálnak. A FIP-jelöltek lokalizációját Gentle AdaBoost technikával javított gyenge osztályozók 10 szintes kaszkádjával, míg a FIP-jelöltek megtartását és elvetését geometriai megszorításokkal végezzük más jelöltekkel vett távolságok és bezárt szögek függvényében.

Míg a Haar jellemzők nagy népszerűségnek örvendenek az arcfelismerésben, a FIP-ek tanulása nehezebb feladat. Egy arcnak több, jobban elkülöníthető jellemzője tanítható. A tanítható jellemzők számának növelése érdekében az osztályozónkat nem csak FIP-ekre, hanem az egész kódra is taníthatjuk. Az adatrégió változatossága ellenére jobb eredményt érhetünk így el, amit segítenek az adatsűrűségi minták és a negyedik, orientációs FIP tanulása. (3(b) ábra)

Az LBP-re és HOG-ra alapuló osztályozók szintén taníthatóak FIP-re és teljes kódra, és mivel a kiértékelési sebességben nem különböznek jelentősen a Haar jellemzőn alapuló osztályozóktól, alkalmasak valós idejű lokalizálásra. Továbbá az LBP-alapú osztályozók alkalmasabbak is lehetnek ebben az esetben a Haar jellemzőkre alapulóknál, mivel itt nem korlátozódik a tanulás egy előre definiált jellemzőhalmazra. A HOG-alapú osztályozók szintén használható megoldást kínálnak a jellemzők megragadására, mivel a QR kód kötött vizuális struktúrája limitálja az előforduló irányokat.



3. ábra. (a): FIP egy Haar jellemző két példányával. A jellemző a belső és külső fekete régióra is illeszkedik minden irányból, viszont ez az egyetlen minta, ami képes a FIP-re illeszkedni. (b): Példa Haar-jellemzők, melyek illeszkednek a teljes kód régióján belül a FIP-ekre, az orientációs mintára és az adatsűrűségi mintára.

## 2.3. Az osztályozó tanítása

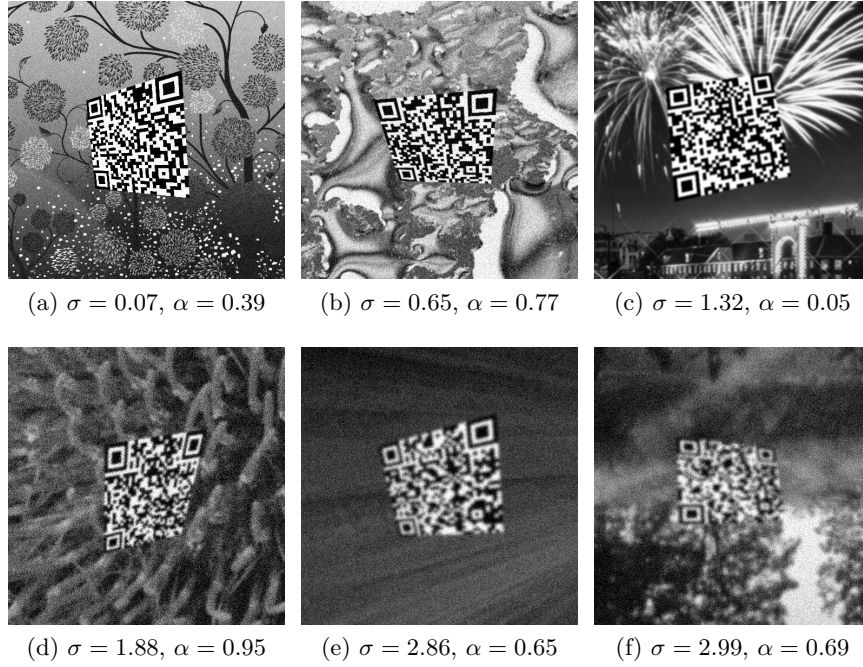
A tanításhoz szükséges egy címkézett előfordulásokat tartalmazó képhalmaz, legalább ezres nagyságrendű példával. Az adatbázis állhat manuálisan készített és

címkézett képekből, viszont ez maga után vonja a címkézés pixelszintű pontatlanságait. Egy több ezer példányos, pontosan címkézett adatbázist hoztunk létre a példányok negatív, objektumot nem tartalmazó képekre vetítésével, mesterségesen transzformálva és zajosítva.

Az osztályozók tanítását az *OpenCV* segítségével végeztük, mely biztosította a jól paraméterezhető osztályozó-tanítást, a Viola-Jones Haar jellemzőhalmazt a Lienhart által kiterjesztett elforgatott entitásokkal, továbbá LBP és HOG alapú tréninget. A jellemzők esetleges szimmetriája, a kaszkád vagy fa topológia, a pozitív minták mérete, a szintek száma, a vágások száma és a teljesítményi mérőszámok mind testreszabhatók.

Az alapértelmezett TPR érték a gyenge osztályozóhoz 0.995-nek, a FPR 0.5-nek lett megválasztva, ami azt jelenti, hogy a minták 99.5 %-a kerül helyesen tovább a következő szintre. Minden szinten egyszerű jellemzők sokaságát használtuk Gentle AdaBoost technikával. Az egyes szinteken álló osztályozók döntési fába is szervezhetők, de nem növelik ebben az esetben a végső hatékonyságot [5]. A szintek számát 10-ben határoztuk meg. Az első négy szinten egyetlen jellemző segítségével sikerült elérni a megfelelő TPR és FPR értékeket, míg a későbbi szinteken több jellemzőre volt szükség, 9 és 15 jellemző között szintenként. A tanítás nem tartalmazott a priori információt arról, milyen jellemzőket részesítsen előnyben.

Összesen hat különböző osztályozót tanítottunk, Haar jellemzőkre, LBP-re és HOG-ra alapozva, mindegyiket a FIP-ekre és az egész kód területére külön tanítva. A FIP-ek esetében a jellemzők szimmetriájának megengedése gyorsítja a tanítást, míg az elforgatott jellemzőhalmaz használata nem növeli jelentősen a hatékonyságot. Ezek a jellemzők nem annyira rugalmasak, hogy egyaránt detektálják a közel egyenes állású és elforgatott kódokat egy osztályozón belül, ennek áthidalására két vagy több külön osztályozó tanítható az elforgatott mintákra, illetve az input kép is elforgatható szükség esetén. A használt mintaméret  $32 \times 32$  volt, ami nagyobb mint a referencia-módszerben. Ezt azért választottuk, mert a teljes kód megtanulása finomabb részletek meglétét igényli. A referencia-módszerben vizsgált kaszkád topológiát választottuk fa helyett, mivel ez magasabb recall-t eredményezett [5]. A mesterségesen generált adatbázis 10 000 képet tartalmaz, melyeken mesterségesen készült QR kódok láthatók, alfanumerikus kódot tárolva. A transzformációk kiválasztásánál azokat a példányokat tartottuk csak meg, melyek egy FIP-szélességnél nem tolják messzebbre a FIP-ek középpontját. Erre azért volt szükség, hogy toleranciákat tudjunk meghatározni a FIP alapú osztályozók utófeldolgozásában a FIP-hármasok alkotására. A képeken ezután Gauss-simítást végeztünk  $[0,3]$  intervallumba eső  $\sigma$  értékkel a kernelben. A képeket ezután zajjal láttuk el, ami egy véletlen zajkép generálását  $I_n$  és annak az eredeti képpel  $I_o$  való súlyozott összeadását jelenti  $I = \alpha I_n + (1 - \alpha) I_o$ , ahol  $\alpha$  a  $[0,0.5]$  tartományba esik. A végleges értékek meghatározásakor szaturációs aritmetikát követve  $[0,255]$  tartományba kényszerítettük a túllógó értékeket. Néhány képet a 4. ábra szemléltet.



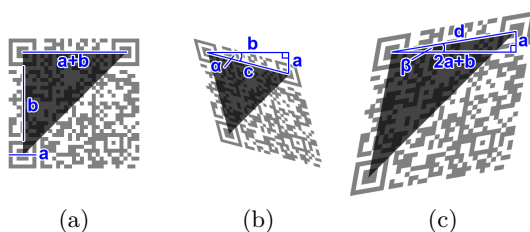
4. ábra. Példák a mesterséges teszt-adatbázisból különböző mértékű simítással és zajjal.

#### 2.4. Utófeldolgozás

A FIP-re tanított osztályozók esetén utófeldolgozásra van szükség, hogy a megtalált FIP-ekből kód-jelölteket állítsunk elő, illetve csökkentsük a hamis entitások számát. Belussi javaslata, hogy menjünk végig az összes lehetséges FIP-en, és alkalmazzunk rájuk geometriai megszorításokat. Ehhez feltételeket kell adnunk az előforduló kódok geometriájáról, a FIP-ek lehetséges távolságáról és az egymással bezárt szögről. A mi esetünkben a kódok 62 byte adatot tárolnak, ami 33:7 Kód:FIP arányt eredményez (5(a) ábra). Eltérő mennyiségű kódolt adat esetében más altípusú QR kód képződik, amire külön osztályozót kell tanítani, illetve átfogalmazni a geometriai feltételeket.

Legyen  $a$  a FIP szélessége és  $b$  a két másik FIP külső éléhez mért távolság. Ha egy kód nem szenved torzulást,  $a+b$  a bal felső FIP távolsága a másik kettőhöz, és a bezárt szög 90 fok (5(a) ábra). Egy QR kód ami akkora torzulást szenved, hogy a FIP-jét maximálisan  $a$  távolsággal tolja befelé (5(b). ábra), detektálható  $T_d = c/(a+b)$  toleranciával a FIP-ek távolságára, ahol  $c = \sqrt{a^2 + b^2}$ . Ha  $(a+b) : a = 33 : 7$  aránnyal számolunk, a képlet 0.7788-at ad  $T_d$ -re, amely 22.12 % toleranciát jelent a várható távolságra. A bezárt szög várhatóan  $90 \pm 20.22^\circ$ , ami a  $T_a = \tan^{-1}(a/b)/90$  képletből számolható, ami 22.47 % tolerancia. A másik esetben, amikor a FIP-et kifelé toljuk (5(c). ábra), hasonló számolással a  $T_d = 0.7707$  és  $T_a = 0.1331$  eredményre jutunk. Ezek alapján a FIP-jelöltek távolságára és

bezárt szögére nagyjából 23 % toleranciát kell hagynunk, hogy ne dobjon el az utófeldolgozási lépés értékes jelölteket. A referencia-algoritmusnál ez a tolerancia 25 %. Ha ezek a FIP-ek különböző méretűek, egy újabb feltétel adható meg a hármasok képzéséhez, ami a FIP-méret toleranciája a hármason belül. Sajnos a FIP méretének perspektív torzulás miatt elszenvedett variabilitása kizárja, hogy értelmesen szűkítsük a jelöltek halmazát ebben a lépésben, a feltétel csak pozitív jelöltek elvesztését eredményezi. Továbbá már a tárgyalt, relatíve kis mértékű torzulások esetén is kellően nagy toleranciákat kell adnunk, melyek erősen limitálják a FIP-hármasok előállításának hatékonyságát.



5. ábra. FIP hármasok toleranciáinak meghatározása. A bal felső FIP helyéből tökéletes esetben 90 fokban  $a + b$  távolságra látszik a két másik FIP. (a); A teszt adatbázisban a FIP-ek befelé (b) és kifelé (c) mozdulhatnak  $a$  nagysággal. Ezen szabályokhoz számolt geometriai toleranciák nem tudják hatékonyan szűrni a nagy számú jelöltet az értékes jelöltek megtartása mellett

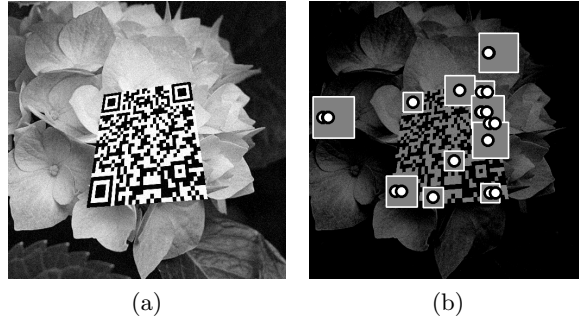
Míg azok az osztályozók, melyek az egész kódon tanultak, nem igényelnek utófeldolgozást, a FIP-re tanultak igénylik a távolságmátrix elkészítését minden FIP-pár között, és egy iránymátrixot ami a közbezárt szögeket tartalmazza. Ezután át kell néznünk az  $n$  jelöltet, és megpróbálni hármasokat alkotni, ami  $\mathcal{O}(n^3)$  időt vesz igénybe. Ez jelentősen csökkenti a végrehajtási sebességet, mivel a FIP-re tanított osztályozó jelentős mennyiségű hamis FIP jelöltet állít (6. ábra).

### 3. Eredmények

A Haar jellemzőket tanuló osztályozó tanítása 15 órát vett igénybe Core 2 Duo 3.00 GHz CPU-n, míg az LBP 1.5 órát, a HOG jellemzők tanulása 30 percig tartott.

A kiértékelést tekintve a különböző osztályozók nem mutattak jelentős teljesítménybeli eltérést, mindegyik alkalmas valós idejű feldolgozásra. A multi-scale feldolgozás csak a skálázási paraméterétől függ. 1.1-es skálátényezővel a feldolgozás kb. 100–200 ms időt vesz igénybe egy  $512 \times 512$  px méretű kép esetében.

Az 1. táblázat tartalmazza az osztályozók teljesítményének mérőszámait. A referencia-algoritmus a HAAR-FIP, 90 % fölötti recall-al rendelkezik, ami megerősíti az eredeti szerzők mérését, és jó alapot szolgál a FIP tanulás hatékony-



6. ábra. QR kód és FIP-re tanított Haar-osztályozó. Eredeti kép (a); jellemzőkép (b) nagy számú FIP jelöléssel (szürke téglalap), és jelölt példányok (fehér kör), melyek átmentek az utófeldolgozási lépésen. A FIP-re tanított osztályozók magas FPR értékkel működnek. Egy jelölt több hármásban is szerepelhet, melyet az átlapoló körök jelölnek.

ságának kiértékelésére. A FIP-re tanított osztályozók pontossága alatta marad a teljes kódon tanult osztályozókénak, ráadásul utófeldolgozást igényel. Az LBP és HOG alapú osztályozók nem rendelkeznek olyan magas recall értékkel, mint a FIP-alapúak, de nem is generálnak akkora mennyiségű hamis pozitív entitást.

1. táblázat. A javasolt osztályozók teljesítmény-mérőszámai. A használt jellemzők a Haar, LBP és HOG. Mindegyik típust külön tanítottunk FIP-re és teljes kód-objektumra is.

	Precision	Hit rate	F-measure
HAAR-FIP [5]	$0.1535 \pm 0.0920$	$0.9436 \pm 0.0753$	$0.2640 \pm 0.1125$
LBP-FIP	$0.1686 \pm 0.0530$	$0.7356 \pm 0.1112$	$0.2743 \pm 0.0773$
HOG-FIP	$0.4753 \pm 0.2466$	$0.7885 \pm 0.1960$	$0.5931 \pm 0.1947$
HAAR-FULL	$0.4208 \pm 0.2404$	$0.9995 \pm 0.1092$	$0.5923 \pm 0.1050$
LBP-FULL	$0.9050 \pm 0.1312$	$0.9999 \pm 0.0857$	$0.9501 \pm 0.0721$
HOG-FULL	$0.5390 \pm 0.2549$	$0.9975 \pm 0.1001$	$0.6999 \pm 0.1221$

Azok az osztályozók, melyek a teljes kódra tanulnak, sokkal jobban teljesítettek. A HAAR-FULL, az LBP-FULL és a HOG-FULL kiemelkedő recallt és elfogadható pontosságot mutatnak. Az LBP-FULL osztályozó képes volt minden kódot megtalálni a teszt adatbázisban alacsony arányú FPR mellett, az F-score értéke 0.95.

A 2. táblázat mutatja Sörös [12] publikus adatbázisán végzett kiértékeléseket. HAAR-FULL, LBP-FULL és HOG-FULL ugyanazok az osztályozók, amik a mesterséges adatbázison tanultak, módosítás nélkül kerültek kiértékelésre. A következő három osztályozó, HAAR-SOROS, LBP-SOROS és HOG-SOROS teljes kódokon tanultak a 100 képből álló publikus adatbázison. Az adatbázis iPhone



kamerával készült, változatos méretű kódokat tartalmaz változatos mértékű simítást elszenvedett képeken. A várakozásnak megfelelően az első három osztályozó alacsonyabb recall-al teljesített, mivel másik adatbázison tanultak más feltételekkel, ettől függetlenül az eredmények bizonyítják, hogy az egyszerű jellemzőket használó kaszkádolt osztályozók használható megközelítései a QR kódok lokalizálásának.

Az utóbbi három osztályozó, melyet a publikus adatbázison tanítottunk, csak 85 példát kapott tanuláshoz és 21-et kiértékeléshez, ami nagyon kevés ahhoz, hogy erős következtetéseket vonjunk le gépi tanulási témában. HAAR-SOROS és HOG-SOROS nem termeltek hamispozitív eredményeket, de képtelenek voltak minden egyedet megtalálni. Az eredmények alapján az LBP a legjobb választás a feladatra, de a valószínűleg túl kevés tanító példa miatt ennek teljesítménye sem kiemelkedő.

2. táblázat. Az osztályozók teljesítménye a Sörös adatbázison [12]. A –FULL-ra végződő osztályozók ugyanazok, amik a szintetikus adatbázison tanultak, míg a –SOROS osztályozók a publikus, 97 képet tartalmazó adatbázishoz készültek.

	Precision	Hit rate	F-measure
HAAR-FULL	$0.2366 \pm 0.2325$	$0.9060 \pm 0.2192$	$0.3752 \pm 0.1285$
LBP-FULL	$0.3663 \pm 0.3265$	$0.7607 \pm 0.1847$	$0.4944 \pm 0.1430$
HOG-FULL	$0.7817 \pm 0.2842$	$0.9487 \pm 0.2871$	$0.8571 \pm 0.2141$
HAAR-SOROS	$0.9999 \pm 0.4220$	$0.7619 \pm 0.2587$	$0.8649 \pm 0.2937$
LBP-SOROS	$0.3684 \pm 0.2082$	$0.9999 \pm 0.1640$	$0.5385 \pm 0.0973$
HOG-SOROS	$0.9999 \pm 0.2127$	$0.9524 \pm 0.1063$	$0.9756 \pm 0.1347$

A leghatékonyabb osztályozó a következő paraméterekkel rendelkezik: LBP jellemzőt használ  $32 \times 32$  mintamérettel, 10 szintes kaszkád topológiába szervezett gyenge osztályozókkal, szintenként Gentle AdaBoost technikával javítva, 0.995 TPR és 0.5 FPR korlátokkal, magasabb rendű struktúra (pl. fa) nélkül. Azokban a végfelhasználói alkalmazásokban, ahol nagyobb az orientáció variabilitása, több különböző orientációra tanított osztályozó együttes munkája szükséges.

## 4. Összefoglalás

A QR kód formátuma jelentősen elterjedt az elmúlt években és az alkalmazások automatikus leolvasást igényelnek. Bemutattunk erre több, gyenge jellemzőket használó kaszkád osztályozót, különböző típusú jellemzőket és bemeneti adatokat használva. Az elkészült osztályozók képesek valós időben nagy pontossággal lokalizálni a QR kódokat a képeken. A paraméterek hangolása az alkalmazások igényeihez szabható.

A vizsgálatok szerint a kaszkád osztályozók alkalmasak a lokalizálási feladatra, különösen az LBP-t használó, teljes kódton tanult osztályozók bírnak jó teljesítménnyel.

## Köszönetnyilvánítás

A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg. Projekt megnevezés: Telemedicina fókuszú kutatások Orvosi, Matematikai és Informatikai tudományterületeken (TOMI). Projektazonosító: TÁMOP-4.2.2.A-11/1/KONV-2012-0073.

## Hivatkozások

1. Parikh, D., Jancke, G.: Localization and segmentation of a 2D high capacity color barcode. In: Applications of Computer Vision, 2008. WACV 2008. IEEE Workshop on. (2008) 1–6
2. Lin, J.A., Fuh, C.S.: 2D barcode image decoding. Mathematical Problems in Engineering (2013)
3. Chu, C.H., Yang, D.N., Pan, Y.L., Chen, M.S.: Stabilization and extraction of 2D barcodes for camera phones. Multimedia Systems **17** (2011) 113–133
4. Ohbuchi, E., Hanaizumi, H., Hock, L.A.: Barcode readers using the camera device in mobile phones. In: Cyberworlds, 2004 International Conference on. (2004) 260–265
5. Belussi, L.F.F., Hirata, N.S.T.: Fast QR code detection in arbitrarily acquired images. In: Graphics, Patterns and Images (Sibgrapi), 2011 24th SIBGRAPI Conference on. (2011) 281–288
6. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. Volume 1. (2001) I–511–I–518 vol.1
7. Lienhart, R., Kuranov, A., Pisarevsky, V.: Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In: Pattern Recognition. Volume 2781 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2003) 297–304
8. Bodnár, P., Nyúl, L.G.: A novel method for barcode localization in image domain. In: Image Analysis and Recognition. Volume 7950 of Lecture Notes in Computer Science. (2013) 189–196
9. Ojala, T., Pietikainen, M., Harwood, D.: Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision and Image Processing., Proceedings of the 12th IAPR International Conference on. Volume 1. (1994) 582–585 vol.1
10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. Volume 1. (2005) 886–893 vol. 1
11. Wang, X., Han, T., Yan, S.: An HOG-LBP human detector with partial occlusion handling. In: Computer Vision, 2009 IEEE 12th International Conference on. (2009) 32–39
12. Sörös, G., Flörkemeier, C.: Blur-resistant joint 1D and 2D barcode localization for smartphones. In: Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia. MUM '13, New York, NY, USA, ACM (2013) 11:1–11:8